Data Intensive Computing on the Grid: Architecture & Technologies

Presented by: Ian Foster

Mathematics and Computer Science Division
Argonne National Laboratory
and
Department of Computer Science
The University of Chicago

Joint work with Ann Chervenak, Carl Kesselman, Steven Tuecke, and others (see acknowledgments)

Overview

- Problem statement
- Grid architecture
- Emerging production Grids
- Our view of data-intensive Grid architecture
- Globus project focus and contributions
- Future directions: The GriPhyN project

Problem Statement

The Problem

"Enable a geographically distributed community [of thousands] to perform sophisticated, computationally intensive analyses on Petabytes of data"

the globus project
www.globus.org

Example Application Scenarios

- Climate community
  - Sharing, remote access to and analysis of Terascale climate model datasets
- GriPhyN (Grid Physics Network)
  - Petascale Virtual Data Grids (see later)
- Distance visualization
  - Remote navigation through large datasets, with local and/or remote computing

the globus project
www.globus.org

Data Intensive Issues Include …

- Harness [potentially large numbers of] data, storage, network resources located in distinct administrative domains
- Respect local and global policies governing what can be used for what
- Schedule resources efficiently, again subject to local and global constraints
- Achieve high performance, with respect to both speed and reliability
- Catalog software and virtual data

Q: Are these issues unique to "data grids"?

the globus project
www.globus.org

Data Intensive Computing and Grids

- The term "Data Grid" is often used
  - Unfortunate as it implies a distinct infrastructure, which it isn't; but easy to say
- Data-intensive computing shares numerous requirements with collaboration, instrumentation, computation, …
- Important to exploit commonalities as very unlikely that multiple infrastructures can be maintained
- Fortunately this seems easy to do!

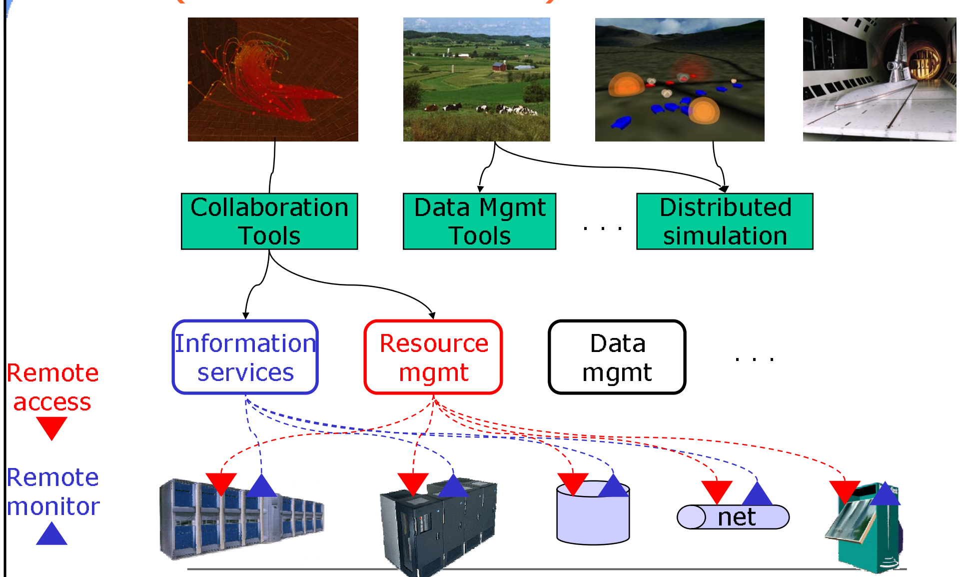
the globus project
www.globus.org

Grid Architecture

Grid Protocols, Services, Tools: Enabling Sharing in Virtual Organizations

- Protocol-mediated access to resources
  - Mask local heterogeneities
  - Extensible to allow for advanced features
  - Negotiate multi-domain security, policy
  - "Grid-enabled" resources speak protocols
  - Multiple implementations are possible
- Broad deployment of protocols facilitates creation of Services that provide integrated view of distributed resources
- Tools use protocols and services to enable specific classes of applications

The Role of Grid Services (aka Middleware) and Tools



the globus project
www.globus.org

Current State of Grid Protocols and Services

- Three basic Grid protocols:
  - GRAM protocol for remote resource mgmt
  - LDAP for resource characterization
  - Remote storage management protocol
- All three leverage Grid Security Infrastructure
- Services include
  - LDAP-based index servers for discovery
  - Brokers for resource management
  - Network Weather Service for network performance
  - Certificate Authorities for credentials

the globus project
www.globus.org

Grid Security Infrastructure (GSI)

- Authentication and authorization for Grids
  - Single-sign on, run anywhere [if authorized]
  - PKI, X.509 certificates
  - Identity/credential mapping at each resource
  - Allows programs to act as user for limited period: delegation of rights
  - Generic Security Services (GSS) API
  - Generic Authorization and Access control (GAA) API call-outs for authorization
- Used in FTP, Condor, SRB, SSH, GRAM, …

Resource Management Protocol (GRAM)

- Remote management protocol for computing resources in Grid environments
  - Layered on HTTP protocol
  - GSI for authentication, authorization
  - Job submission, monitoring, control
  - Executable staging
  - Scheduling, advance reservation, policy, queue time estimation as examples of extended functionality
- Gateways to Condor, LSF, SSH, PBS, others
  - Plus "glide-in" capability for Condor integration

Information Service (MDS aka GIS)

- Resource characterization and discovery services for Grid environments
  - Uses LDAP protocol for data retrieval
  - Integration of variety of information providers
  - Flexible architecture enables creation of application-specific index servers to support range of resource discovery strategies
  - GSI for authentication, authorization (soon)
- Gateways to GRAM, site, NWS data
  - Plus storage information (soon)

Examples of Tools

- MPICH-G2 for distributed computation
  - Allows arbitrary Message Passing Interface programs to run in heterogeneous systems
- Condor-G as a job submission system
  - Supports submission, monitoring, control of collections of tasks and task graphs
- Java CoG Kit [Commodity Grid Toolkit] for Portal development and access from Java
  - Java interfaces to Grid protocols & services, middle tier services for Web-based portals, GUI elements

Emerging Production Grids

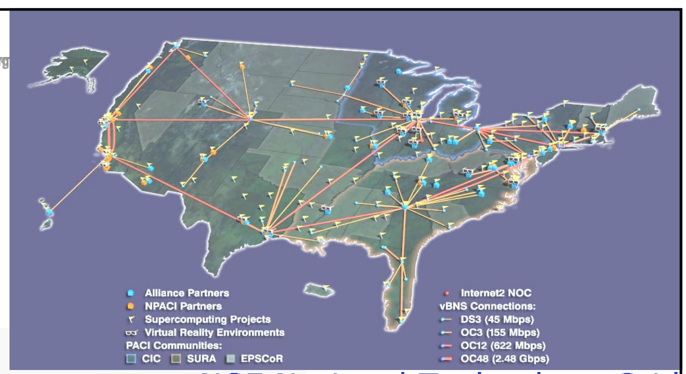
the globus project
www.globus.org

Emerging Production Grids

- A production Grid is one in which Grid protocols and services are deployed across a set of resources in a fashion that is
  - Persistent
  - Supported by computer systems staff
- Obviously a continuum with respect to how much support, what services and tools, scale
- Note that many such Grids can and will exist, serving different communities
  - Standard protocols enable interoperation

the globus project
www.globus.org

Emerging Production Grids (U.S.)



NSF National Technology Grid

+ DOE ASCI DISCOM, DOE Science Grid, others

NASA Information Power Grid

National Technology Grid

- In principle, a single infrastructure; in practice, two distinct efforts for now
- Both PACIs have been running Grid services for several years in "experimental" mode
- Slow progress towards "production"
  - Alliance "Virtual Machine Room"
  - NPACI: HotPage Portal now in production use
- Significant contributions from both Alliance and NPACI to Grid development

Details

- Alliance Virtual Machine Room
  - Focus: remote access to major resources
  - GSI-FTP, GSI-SSH, GRAM
  - MDS in place but not yet production
  - VMR allocation and support mechanisms
  - Production Certificate Authority
- NPACI
  - GSI-FTP for HPSS, GRAM deployed widely
  - Production applications running regularly, HotPage portal
  - Production CA, automatic setup processes

the globus project
www.globus.org

NASA Information Power Grid

- Ames, Glenn, Goddard, Langley
- Full deployment of Grid services
  - Major emphasis on reliability and support
  - Innovative work on advance reservation
  - So far major resources not included

the globus project
www.globus.org

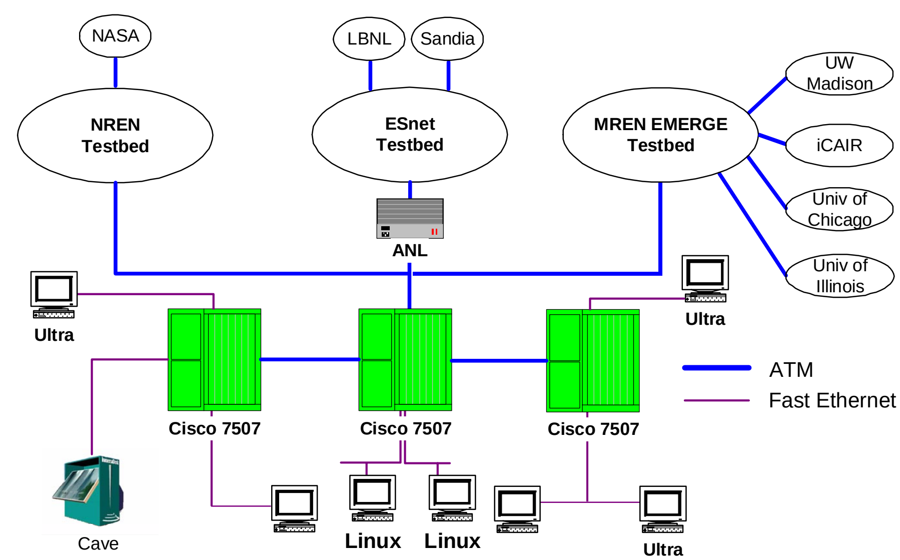
DISCOM (DOE ASCI)

- Aggressive and well-managed program to create tri-lab integrated computing env.
- On track to complete deployment by end of FY2000
- Kerberos rather than PKI for security
  - Easy due to use of GSS-API for all Grid security
- Innovative development work relating to resource brokers

DOE Science Grid

- Goal: Deploy Grid Services across a set of DOE Science labs and partner institutions
- Early work conducted during DOE NGI program (QoS testbed: see next slide)
- Now lobbying MICS for funding
  - E.g., for DOE Science Certificate Authority

The GARNET Testbed: A Contributor to the DOE Science Grid



Production Deployment: Summary

- Significant progress has been made towards "production" Grid deployment
- Has proved more difficult than anticipated
  - Deployment uncovered new problems (e.g., certificate authority operations)
  - Lack of resources: all shoe-string operations
  - New skills required in staff
- No show-stoppers uncovered
- Challenge remains: reducing cost of entry

Architecture and Services for Data Intensive Grid Computing

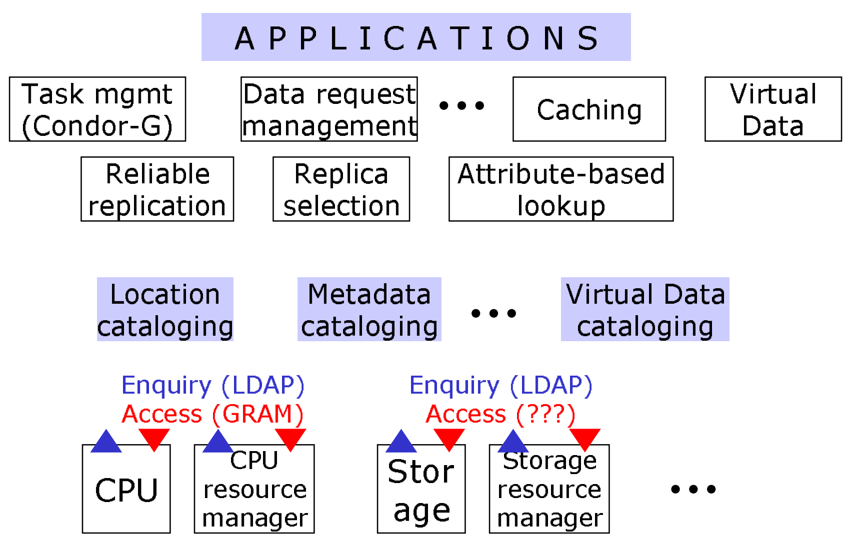
Examples of Desired Data Grid Functionality

- High-speed, reliable access to remote data
- Automated discovery of “best” copy of data
- Manage replication to improve performance
- Co-schedule compute, storage, network
- “Transparency” wrt delivered performance
- Enforce access control on data
- Allow representation of “global” resource allocation policies

Central Q: How must Grid architecture be extended to support these functions?

the globus project
www.globus.org

“Data Grid” Architecture Elements



the globus project
www.globus.org

Examples of Grid Services

- Grid-enabled storage
  - Authenticated, cross-domain transport
  - Storage system characterization
- Location cataloging
  - Records location of data elements (files)
  - Also provides access to relevant data e.g. performance of associated storage systems
- Metadata cataloging
  - Attribute-to-data element mapping

the globus project
www.globus.org

Examples of Tools

- Replica selection
  - Select “best” replica according to e.g. speed
- Reliable replication
  - Generate new replica and update catalog, dealing with failures
- Request management
  - Schedule and monitor a set of requests, using enquiry and data access protocols
  - C.f. LBNL request manager

the globus project
www.globus.org

Globus Project Contributions to Data Grid Architecture

the globus project
www.globus.org

Globus Project Contributions to Data Grid Architecture

- Low-level protocols and services
  - Grid Security Infrastructure
  - Resource management protocol
  - Resource discovery service
- Additional data-oriented services
  - Common data access protocol
  - Replica catalog service
- Higher-level data-oriented services and tools
- Beta Grid Nodes: Standard software load for standard cluster hardware config, hence defining replicable Grid building block

Discussed above

Globus Project Contributions: Overview

1) Common data access protocol
   - Motivation and approach
   - Family of libraries and tools
2) Replica management services
   - Catalog and replication libraries
3) Case study: Earth System Grid
4) Beta Grid Node concept + software
5) Short-term and medium-term plans

For more details, see: http://www.globus.org/datagrid/deliverables

(1) Motivation for a Common Data Access Protocol

- Existing distributed data storage systems
  - DPSS, HPSS: focus on high-performance access, utilize parallel data transfer, striping
  - DFS: focus on high-volume usage, dataset replication, local caching
  - SRB: connects heterogeneous data collections, provides uniform client interface, metadata queries
- Problem: Incompatible protocols
  - Require custom clients
  - Has the unfortunate effect of partitioning available data sets and storage devices

A Common, Secure, Efficient Data Access Protocol

- Common, *extensible* transfer protocol
- Decouple low-level data transfer mechanisms from the storage service
- Advantages:
  - New, specialized storage systems are automatically compatible with existing systems
  - Existing systems have richer data transfer functionality
- Interface to many storage systems
  - HPSS, DPSS, file systems
  - Plan for SRB integration (see below)

Common Data Access Protocol and Storage Resource Managers

- Grid encompasses "dumb" & "smart" storage
- All support base functionality
  - "Put" and "get" as essential mechanisms
  - Integrated security mechanisms, of course
- Storage Resource Managers can enhance functionality of selected storage systems
  - E.g., progress, reservation, queuing, striping
  - Plays a role exactly analogous to "Compute Resource Manager"
- Common protocol means all can interoperate

the globus project
www.globus.org

A Universal Access/Transport Protocol

- Suite of communication libraries and related tools that support
  - GSI security
  - Third-party transfers
  - Parameter set/negotiate
  - Partial file access
  - Reliability/restart
  - *Logging/audit trail*
  - Integrated instrumentation
  - Parallel transfers
  - Striping (cf DPSS)
  - Policy-based access control
  - *Server-side computation [later]*
- All based on a standard, widely deployed protocol

the globus project
www.globus.org

And the Universal Protocol is … GSI-FTP

- Why FTP?
  - Ubiquity enables interoperation with many commodity tools
  - Already supports many desired features, easily extended to support others
- We use the term GSIFTP to refer to
  - Transfer protocol which meets requirements
  - Family of tools which implement the protocol
- Note GSI-FTP > FTP
- Note that despite name, GSI-FTP is not restricted to file transfer!

GSI-FTP: Basic Approach

- FTP is defined by several IETF RFCs
- Start with most commonly used subset
  - Standard FTP: get/put etc., 3rd-party transfer
- Implement RFC'ed but often unused features
  - GSS binding, extended directory listing, simple restart
- Extend in various ways, while preserving interoperability with existing servers
  - Stripe/parallel data channels, partial file, automatic & manual TCP buffer setting, progress and extended restart

The GSI-FTP Family of Tools

- Patches to existing FTP code
  - GSI-enabled versions of existing FTP client and server, for high-quality production code
- Custom-developed libraries
  - Implement full GSI-FTP protocol, targeting custom use, high-performance
- Custom-developed tools
  - E.g., high-performance striped FTP server

Family of Tools
Patches to Existing Code

- Patches to standard FTP clients and servers
  - gsi-ncftp: Widely used client
  - gsi-wuftpd: Widely used FTP server
  - GSI modified HPSS pftpd
  - GSI modified Unitree ftpd
- Provides high-quality, production ready, FTP clients and servers
- Integration with common mass storage systems
- Do not support the full gsi-ftp protocol

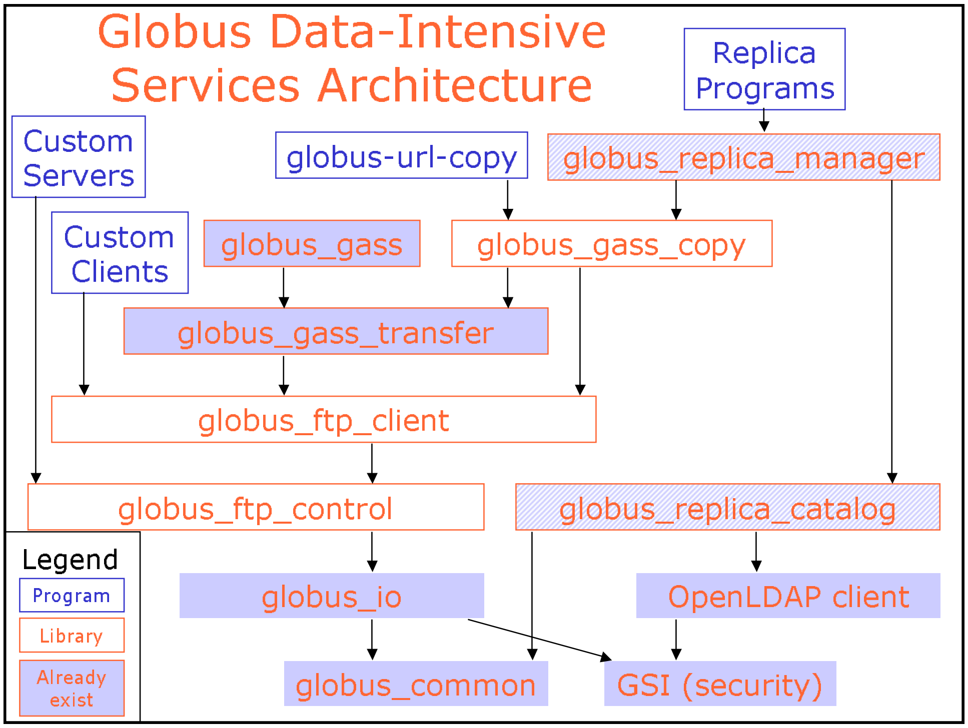
Family of Tools
Custom Developed Libraries

- Custom developed libraries
  - globus_ftp_control: Low level FTP driver
    - Client & server protocol and connection management
  - globus_ftp_client: Simple, reliable FTP client
  - globus_gass_copy: Simple URL-to-URL copy library, supporting (gsi-)ftp, http(s), file URLs
- Implement full gsi-ftp protocol
- Various levels of libraries, allowing implementation of custom clients and servers
- Tuned for high performance on WAN

Family of Tools
Custom Developed Programs

- Simple production client
  - globus-url-copy: Simply URL-to-URL copy
- Experimental FTP servers
  - Striped FTP server (ala.DPSS)
  - Multi-threaded FTP server with parallel channels
  - Firewall FTP proxy: Securely and efficiently allow transfers through firewalls
  - Striped interface to parallel file systems
- Experimental FTP clients
  - POSIX file interface

Globus Data-Intensive Services Architecture



(2) The Replica Management Problem

- Maintain a mapping between logical names for files and collections and one or more physical locations
- Important for many applications
  - Example: CERN HLT data
    - Multiple petabytes of data per year
    - Copy of everything at CERN (Tier 0)
    - Subsets at national centers (Tier 1)
    - Smaller regional centers (Tier 2)
    - Individual researchers will have copies

Our Approach to Replica Management

- Identify replica cataloging and reliable replication as two fundamental services
  - Layer on other Grid services: GSI, transport, information service
  - Use LDAP as catalog format and protocol, for consistency
  - Use as a building block for other tools
- Advantage
  - These services can be used in a wide variety of situations

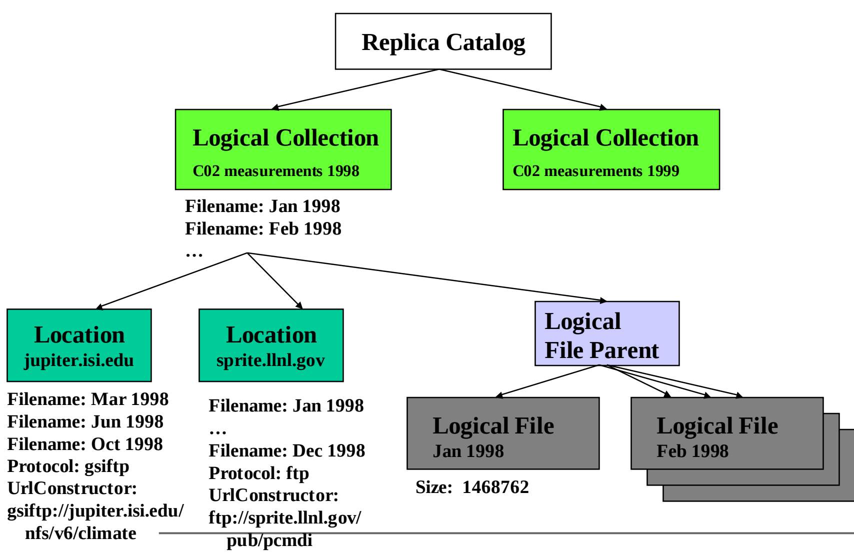
the globus project
www.globus.org

Replica Manager Components

- Replica catalog definition
  - LDAP object classes for representing logical-to-physical mappings in an LDAP catalog
- Low-level replica catalog API
  - globus_replica_catalog library
  - Manipulates replica catalog: add, delete, etc.
- High-level reliable replication API
  - globus_replica_manager library
  - Combines calls to file transfer operations and calls to low-level API functions: create, destroy, etc.

the globus project
www.globus.org

Replica Catalog Structure: A Climate Modeling Example



Replica Catalog API

- globus_replica_catalog_collection_create()
  - Create a new logical collection
- globus_replica_catalog_collection_open()
  - Open a connection to an existing collection
- globus_replica_catalog_location_create()
  - Create a new location (replica) of a complete or partial logical collection
- globus_replica_catalog_fileobject_create()
  - Create a logical file object within a logical collection

Replica Catalog API (cont.)

- globus_replica_catalog_collection_list_filenames()
  - List all logical files in a collection
- globus_replica_catalog_location_search_filenames()
  - Search for the locations (replicas) that contain a copy of all the specified files
- globus_replica_catalog_location_delete()
  - Delete a location (replica) from the catalog
- globus_replica_catalog_collection_delete()
  - Delete a logical collection from the catalog

the globus project
www.globus.org

Reliable Replication API

- Globus_replica_management_copy_files()
  - Copy a set of filenames from one physical location to another physical location
  - Update the replica catalog to add new location object (if necessary)
  - Add new filenames to location object
- Globus_replica_management_synchronize_filenames()
  - Ensure that the location object for a physical storage directory correctly reflects the contents of the directory

the globus project
www.globus.org

Replica Catalog Services as Building Blocks: Examples

- Combine with information service to build <u>replica selection</u> services
  - E.g. "find <u>best</u> replica" via NWS data
  - Use of LDAP as common protocol for info and replica services makes this easier
- Combine with application managers to build <u>data distribution</u> services
  - E.g., build new replicas in response to frequent accesses

Outstanding Issues for Replica Management

- Current architecture assumes a read-only workload
  - What write consistency should we support?
- What high-level operations are needed?
  - Combine storage and catalog operations
- Support replication in Objectivity DB?
- Replicating the replica catalog
- Replication of partial files
- Alternate catalog views: files belong to more than one logical collection

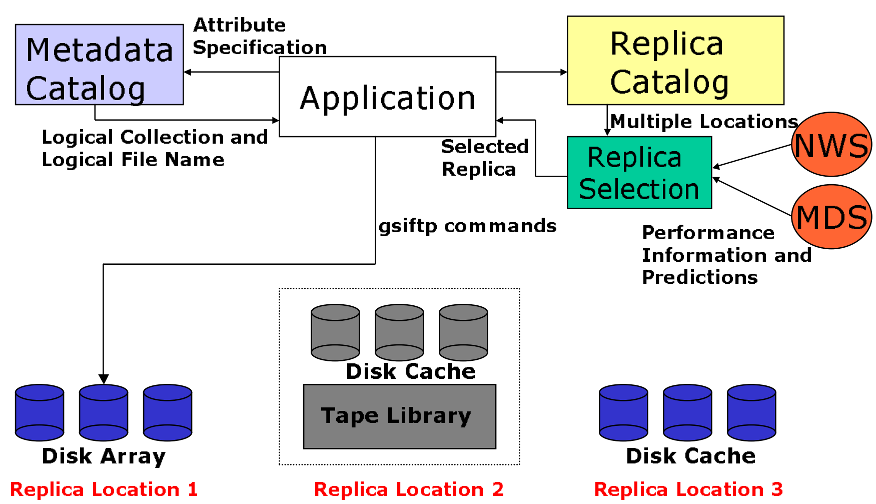
Relationship to Metadata Catalogs

- Must support a variety of metadata catalogs
  - MCAT being one important example
  - Others include LDAP catalogs, HDF
- Two possible approaches to integration
  - Use LDAP to define standard access methods to diverse metadata catalogs (I.e., map metadata formats to LDAP object classes)
  - Define some other standard metadata lookup API/protocol, returning logical collection/file names required by replica catalog

(3) Case Study: Earth Systems Grid

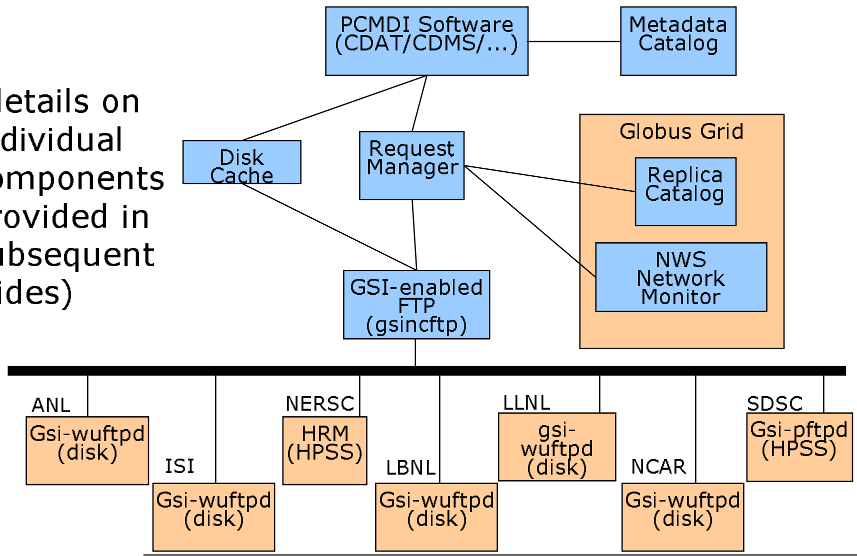
- Goal: enable efficient distributed access to large (TB-scale) climate model data sets
- Approach
  - "Grid-enable" desktop analysis software from PCMDI @ LLNL: transparent access
  - Build on replica management and GSI-FTP software
  - "Request Manager" from LBNL used to coordinate selection and transfer
  - Network Weather Service data used for replica selection

High-Level View of ESG : A Model Architecture for Data Grids

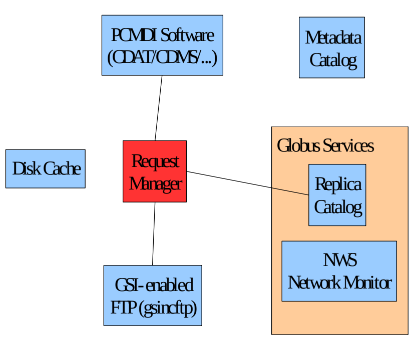


Detailed View of Earth System Grid Scenario

(details on individual components provided in subsequent slides)

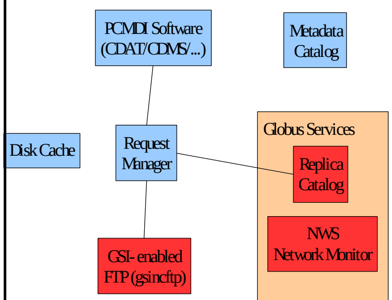


Details (1): Request Manager



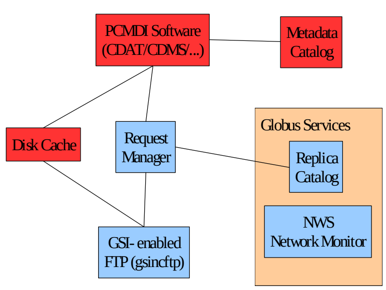
- receives and collates requests for data
- determines locations of data files by querying the Globus replica catalog
- uses the NWS network monitor to determine which location can be accessed most rapidly
- provides time estimates of data transfer completion
- uses gsincftp to transfer data

Details (2): Data Transfer



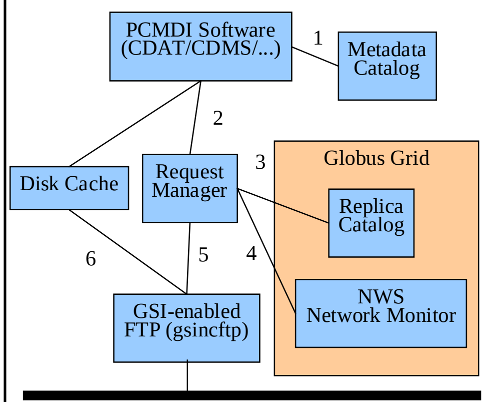
- Replica catalog
  - Catalog stores locations, size of data files, which may be replicated
  - Implemented using LDAP directory: distributed
- GSI-enabled FTP tools
  - Grid Security Infrastructure (public-key)
  - Supports proxy certificates: single sign-on
- NWS
  - Dynamically forecasts network performance

Details (3): PCMDI Software System



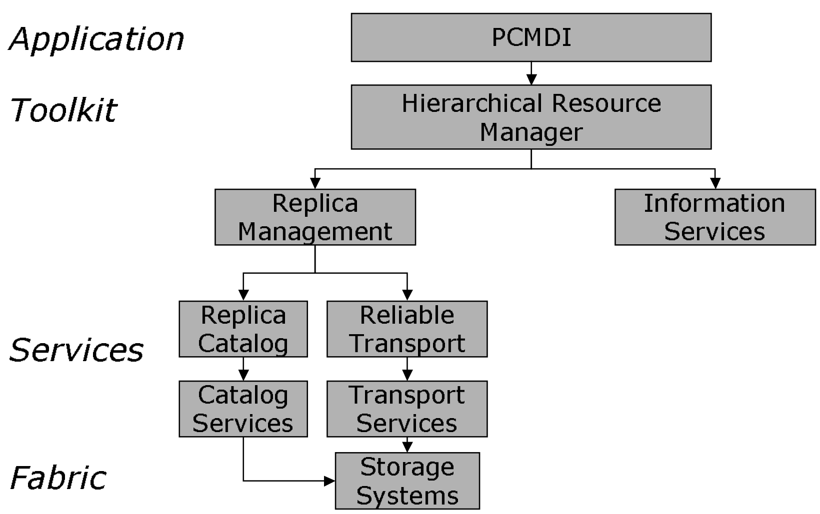
- CDAT analysis tool, CDMS data management system, ...
- CDMS provides a logical view of climate data
  - A dataset is a collection of related files of similar structure, typically the result of one GCM run
  - Datasets may also be viewed as a collection of data variables which can span multiple files
  - A database is a collection of datasets
  - A metadata catalog stores descriptions of databases, datasets and variables. This can be searched to locate data of interest
  - Metadata catalogs can be implemented as LDAP directories (distributed)

Operation: Client-side Data Extraction and Visualization



1) Use PCMDI Software to search a metadata catalog, locate a dataset and variable of interest.
2) Issue a request for data. This may consist of many files or portions of files. Provide a proxy certificate.
3) Search a replica catalog to locate the data. Files may exist in multiple locations.
4) Consult NWS to estimate the fastest network link to a replica.
5) Request the file from a Globus FTP server; authenticate the user with the proxy certificate.
6) Transfer data to a local disk cache

ESG Example from the Perspective of Grid Services Hierarchy



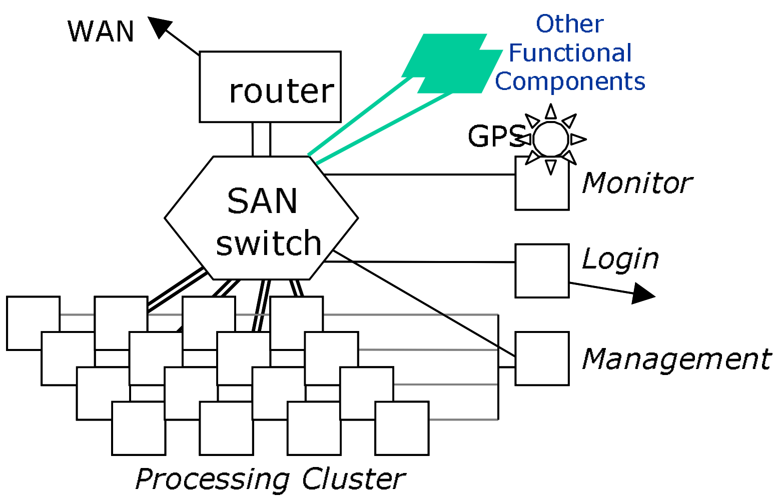
the globus project
www.globus.org

(4) Beta Grid Nodes (BGNs)

- Commodity storage/compute clusters
- Geographical distribution
- Standard software load
  - Grid-enabled support for remote access
  - Data management for caching
  - Scheduling for computation
  - Authentication, authorization, reservation
- Remote management (power etc.)
- High-speed network interconnect
- “Bricks” to be distributed over network

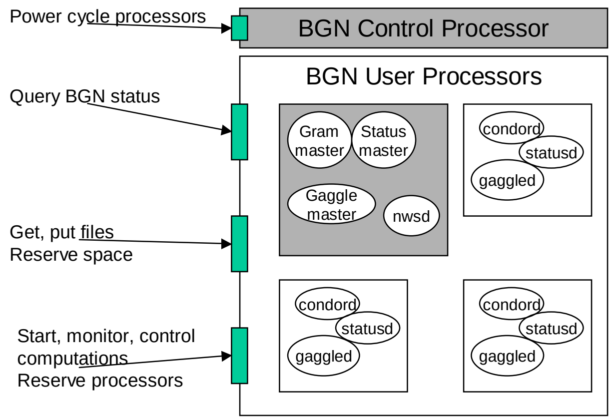
the globus project
www.globus.org

BGN Architecture: Physical



the globus project
www.globus.org

BGN Architecture: Logical



the globus project
www.globus.org

BGN Development Plans

- Phase 1
  - High-speed FTP, information service
  - Dynamic accounts
- Phase 2
  - Condor for CPU scheduling (Phase 1?)
  - Space and CPU reservation
  - Remote management
- Then logging, policy, …
- Collaborative development process
  - ANL, UChicago, NCSA, Indiana, Wisconsin

the globus project
www.globus.org

(5) Globus and Data Grids: Short & Medium-Term Plans

- 6 month
  - Complete bulk of FTP extensions
  - Robust location cataloging services
  - Storage information service
  - Integration with apps and tools: e.g., SRB
  - High-performance striped server (w/ LBNL)
  - Beta Grid Node software release
- 6-18 month
  - Policy-based access control
  - Storage reservation
  - GriPhyN research agenda

the globus project
www.globus.org

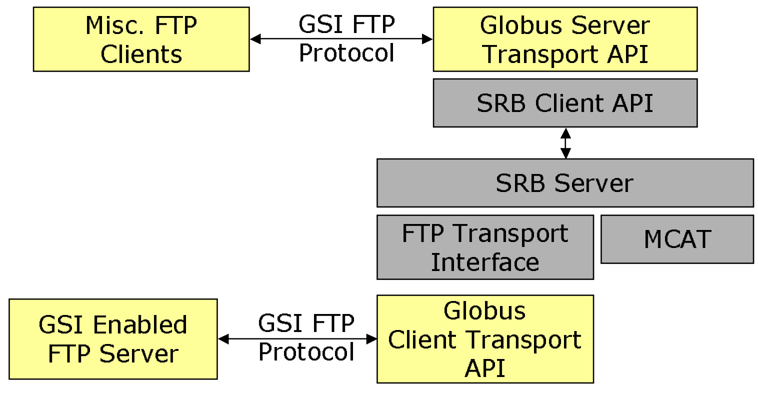
GSI-FTP Availability

- Now
  - Gsi-ncftp, gsi-wuftpd
  - Draft of white paper
  - Draft API and protocol documentation
- August
  - Alpha version of custom libraries and tools
- Fall
  - Globus Toolkit V1.2: Optimized, supported, …

the globus project
www.globus.org

Globus and SRB: Integration Plan

- FTP access to SRB-managed collections
- SRB access to Grid-enabled storage systems



the globus project
www.globus.org

Future Directions: GriPhyN

GriPhyN Overview (www.griphyn.org)

- 5-year, $12.5M NSF ITR proposal to realize the concept of virtual data, via:
  1) CS research on
     > Virtual data technologies (info models, management of virtual data software, etc.)
     > Request planning and scheduling (including policy representation and enforcement)
     > Task execution (including agent computing, fault management, etc.)
  2) Development of Virtual Data Toolkit (VDT)
  3) Applications: ATLAS, CMS, LIGO, SDSS
- PIs=Avery (Florida), Foster (Chicago)

GriPhyN Participants

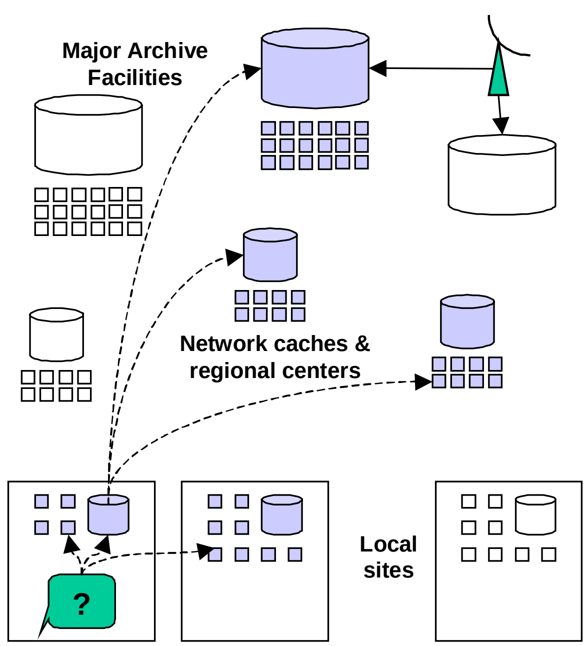
- Computer Science
  - U.Chicago, USC/ISI, UW-Madison, UCSD, UCB, Indiana, Northwestern, Florida
- Toolkit Development
  - U.Chicago, USC/ISI, UW-Madison, Caltech
- Applications
  - ATLAS (Indiana), CMS (Caltech), LIGO (UW-Milwaukee, UT-B, Caltech), SDSS (JHU)
- Unfunded collaborators
  - UIC (STAR-TAP), ANL, LBNL, Harvard, U.Penn

The Petascale Virtual Data Grid (PVDG) Model

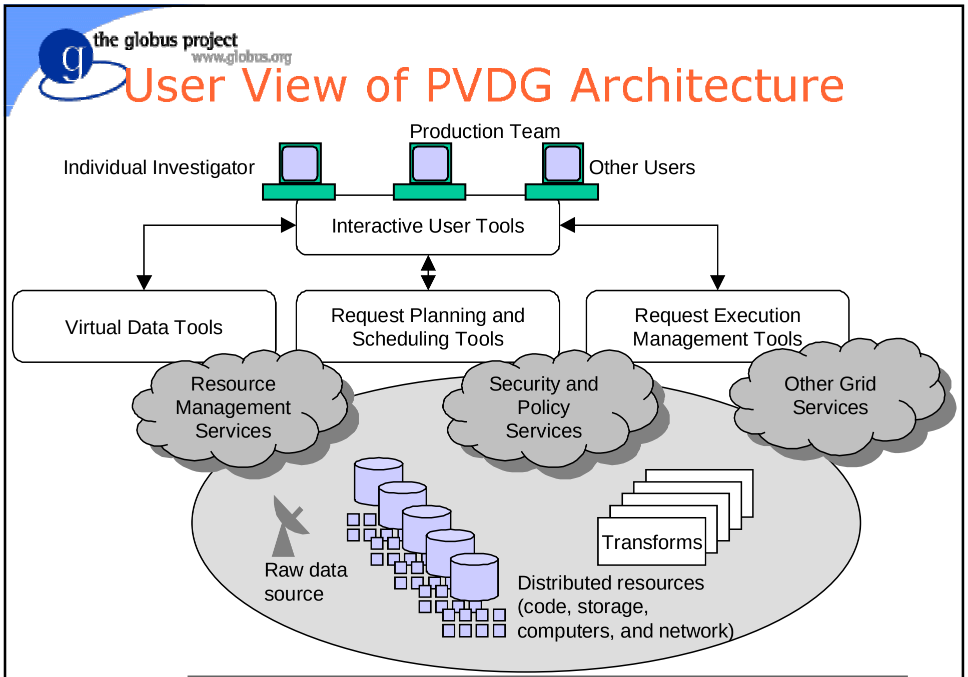
- Data suppliers publish data to the Grid
- Users request raw or derived data from Grid, without needing to know
  - Where data is located
  - Whether data is stored or computed
- User can easily determine
  - What it will cost to obtain data
  - Quality of derived data
- PVDG serves requests efficiently, subject to global and local policy constraints

PVDG Scenario

User requests may be satisfied via a combination of data access and computation at local, regional, and central sites



User View of PVDG Architecture



GriPhyN Milestones (Abbreviated)

	Year 1	Year 2	Year 3	Year 4	Year 5
Virtual Data	Preliminary data models for virtual data.	Procedural / declarative representations	Info. model for catalog planning & scheduling	Integrate perf. estimates into info models	Integrate policy constraints into info models
Request Planning	Model for policy driven request planning	Integration of virtual data and resource models	Request planning algorithms	Integration of local / global policies	Optimization across local & global policies
Request Execution	Language for distributed service	Integration with request planning model	Integration of fault tolerance mechanisms	Support for dynamic replanning	Simulation of Grid behavior
Virtual Data Toolkit	Package basic services	Centralized virtual data services	Distributed virtual data services	Scale virtual data services to Petascale	Enhance usability, performance, etc.
SDSS	Catalog model, data model	Distributed statistics	Distributed virtual data presentation	Distributed generation of merged sky maps	Distributed creation of derived catalogs
LIGO	Catalog, information model	Data streaming through caches	Opportunistic computing of derived products	Event-based derived products	Optimization of product derivation
LHC CMS / ATLAS	Proto Tier-2 system	Replication, caching, and distributed databases	Production prototype for PVDG	Production quality PVDG	Full distributed generation of derived data products

Virtual Data Toolkit

Applns	ATLAS, CMS, LIGO, SDSS, etc., Petascale Virtual Data Grids
	Use to implement specific application-level capabilities
Tools	Planning, estimation, execution, monitoring, curation, etc., etc.
	Provides client access to, integrates to enhance appln-level capabilities
Services	Archive, cache, transport, agent, catalog, security, policy, etc., etc.
	Encapsulates, discovers/publishes/ enhances capabilities of, manages
Fabric	Computer, network, storage, and other resources

GriPhyN and PPDG

- ITR funding gives GriPhyN a rather different focus from PPDG
  - NSF wants innovative CS and solutions to general problems of data-intensive science
- At the same time, effective tech transfer to physics experiments is essential!
  - Yet total budget is small (relative to scope)

=> A close partnership with PPDG is vital to the success of GriPhyN

Defining the GriPhyN/PPDG Relationship

- Important that we define clearly distinct roles
  - E.g., technology areas (PPDG: storage resource managers, high-performance transport; GriPhyN: scheduling, virtual data)
  - E.g., physics experiments (?)
- Methods for consultation and collaboration need to evolve over time, e.g. for
  - Joint development of VDT components
  - Joint application development & experiments
  - Advisory boards

Globus-Related Acknowledgements

- Globus data services design: Ann Chervenak, Carl Kesselman, Steven Tuecke
- FTP: Joe Bester, John Bresnahan, Sam Meder, Von Welch, Wayne Schroeder
- GASS tools: Joe Insley, Stuart Martin
- Replica catalog: Darcy Quesnel, Veronika Nefedova, Steve Fitzgerald, Chuck Salisbury
- PVDG architecture: Miron Livny, Arie Shoshani, Brian Tierney, Reagan Moore, Harvey Newman

the globus project
www.globus.org

The End